Creating Inclusive Developers: Integrating Accessibility Education Through a Practical Learning Method

Alan T. A. Silva[1], **Suzane S. dos Santos**[2], **Marcelle P. Mota**[1]

[1]Instituto de Ciências Exatas e Naturais – Universidade Federal do Pará (UFPA)
Rua Augusto Corrêa, 01 - Guamá, 66075-000, Belém - PA – Brazil

[2]Instituto De Ciências Matemáticas e de Computação – Universidade de São Paulo (USP)
Av. Trab. São Carlense, 400 - Centro, 13566-590, São Carlos - SP – Brazil,

alantas@outlook.com, suzanesantos@usp.br, mpmota@ufpa.br

Abstract. ***Introduction****: Although essential, software accessibility is still absent from many higher education computing programs, leaving a gap in the training of professionals. While past research has proposed methods for its inclusion, these often do not provide the resources needed for use.* ***Objective:*** *This work aims to address this gap by proposing a practical, structured, and reproducible accessibility teaching method to facilitate the inclusion of accessibility in computing courses.* ***Methodology or Steps****: To address this, we present and evaluate an accessibility teaching method that combines Project-Based Learning and Design Thinking. The key contribution is that it provides all necessary materials for its implementation: lesson plans, activity guides, and slides.* ***Results:*** *Results from three applications with students and professors indicate increased awareness and understanding of accessibility and positive feedback. These findings suggest the method can foster empathy and practical skills, supporting the inclusion of accessibility in computing curricula.*
Keywords *Accessibility, Teaching, Human-Computer Interaction, Teaching Method*

1. Introduction

Over the years, several studies have investigated the scenario of accessibility adoption by the software industry [Antonelli et al. 2018, Bi et al. 2022, Freire et al. 2008, Gomes et al. 2020]. The studies show that despite advances, accessibility is still often overlooked. Presenting growth in relation to previous years, but still with very incipient values, with data from 2021 indicating that only around 28% of developers considered that accessibility needs to be incorporated into all software products [Bi et al. 2022].

Studies also sought to understand the main reasons for this scenario [Silva et al. 2022, Teran et al. 2021], identifying that these problems exist from different perspectives. One of these is an educational perspective related to the formal teaching of software accessibility in computing programs, revealing a gap in education on accessibility for these professionals. The studies indicate that accessibility usually does not have dedicated courses and that, in fact, accessibility is hardly presented in the teaching contexts of the programs.

Because of this lack of exposure, these newly graduated professionals already start their careers with a gap in accessibility knowledge, which, along with other problems,

makes it even more difficult for accessibility to be considered in industry projects. And allowing students to graduate without exposure to topics on software accessibility perpetuates "the cycle of ignorance among ICT developers" and the "status quo of exclusion and marginalization of people with disabilities who cannot use the inaccessible products created by the ignorant developers" [Bohman 2012].

Educators have emphasized the need for students to develop awareness and understanding of accessibility [Putnam et al. 2016], and while efforts are being made to improve this adoption of accessibility in software, this educational gap needs to be addressed, and proposing ways to integrate accessibility into the curricula of computing programs can assist with this process. This is essential to guarantee the complete training of these professionals, who will understand more about the technical aspects of accessible development and also be able to show greater empathy toward people who need accessibility by recognizing their challenges and knowing there are ways they can help.

Several studies have proposed methods to support educators and academic programs in integrating accessibility into their curricula [Palan et al. 2017, El-Glaly 2020, Zhao et al. 2020, Ludi et al. 2018]. However, the literature still lacks comprehensive proposals that both demonstrate their effectiveness and provide instructors with the necessary information and resources for straightforward adoption and implementation. Educators themselves often face several challenges when attempting to incorporate accessibility topics into their courses. These include limited institutional support and insufficient access to appropriate teaching materials and training [Putnam et al. 2016].

This study proposes and evaluates an accessibility teaching method to be applied as a portion of computing courses. This method was designed using principles from Project-Based Learning (PBL) and Design Thinking (DT) and aims to be readily adoptable by providing professors with all the necessary class planning, application manuals, and slide presentations. The proposed method was developed to allow for applications with different durations, depending on the time availability. The shortest possible version needs around three hours to be applied, while the longest version can take up to 8 hours, which can be divided between multiple classes.

The text is organized into six more sections: Section 2 presents the related work; Section 3 describes the research method that was applied; Section 4 introduces the proposed accessibility teaching method; Section 5 reports the results; Section 6 discusses the findings; and Section 7 presents the final considerations.

2. Related Work

Several studies, have explored different methods for teaching accessibility, ranging from traditional lectures to hands-on projects and direct interaction with individuals with disabilities. However, there is a lack of shared resources for instructors with less background in the area to help them incorporate these topics into their curricula [Putnam et al. 2016], making it challenging to implement accessibility education more broadly.

A week of accessibility lectures was offered to students in [Palan et al. 2017], and the overall knowledge and awareness of accessibility were measured before and after

the lectures. The results showed that after the lectures, there was an overall increase in these metrics. Despite the positive results presented, the authors do not describe the lecture's content, only the topics of each lecture, making it hard to reproduce. Lectures were also used as part of a method to build accessibility into an introductory course in a Computer Science program in [Gabbert 2020]. As an introductory course, it covers various computing topics, and during lectures, all content was presented connecting it to accessibility. The method was applied twice, but the material used in the classes was not made available for reproduction. Students were encouraged to rate the lessons received, and based on the analysis of the responses, the method was generally well-rated.

Carrying out projects alongside lectures was explored in [El-Glaly 2020], which presented the development of a dedicated accessibility course through lectures and implementation projects offered to Software Engineering students. Through a qualitative analysis of a post-course questionnaire, the author found that the students believed the course helped teach them about accessibility. [Wang 2012] presented a holistic approach to teaching accessibility through lectures and projects. To accomplish this, lectures were presented covering a wide range of topics related to web design, highlighting the accessibility aspects of each one. The students were evaluated to pass the course; however, the method's effectiveness was not evaluated in any way.

First-hand experience with people with disabilities was also used alongside lectures and projects as a method to teach accessibility in [Ludi et al. 2018]. During the three semesters of carrying out the method, students attended lectures, carried out projects, and had direct contact with a person with a disability who could provide feedback regarding their projects. Through the qualitative and statistical analysis of questionnaires answered by 95 students before and after the test application, it was possible to affirm the effectiveness of the proposed method. However, the work does not offer the material used in the classes or details about the projects, which restricts the reproduction of the method.

[Waller et al. 2009] suggested a change in the academic curriculum of computing programs, with accessibility being inserted as a topic in existing courses during the whole program duration. In this work, students were exposed to lectures, experiments, and implementation work. Students were evaluated only within the context of each course, and only the topics that were taught were disclosed, which makes their reproduction challenging. Nothing is directly evaluated as to the effectiveness of the proposed method.

An extensive 4-year study across 29 courses comparing four different methods to teach accessibility: lectures, team projects, direct interaction with someone with a disability, and collaborating with a student with a disability were explored in [Zhao et al. 2020]. Students had their accessibility knowledge and awareness assessed before and after each method, and the results indicated lectures, projects, and interactions as the most effective. The study did not intend to propose a new method to teach accessibility, but to compare traditional methods and present itself as a good resource when choosing between them.

The related studies presented share similarities with the method proposed by this work. Some of them adopted a similar structure, combining lectures and implementation projects for the execution of the proposed methods [El-Glaly 2020, Wang 2012]. Similar

to our approach, other studies also proposed methods to be implemented as a portion within a pre-existing course [Ludi et al. 2018, Palan et al. 2017]. The evaluation of the effectiveness of our method was based on the approach introduced in [Palan et al. 2017], which has also been employed in [Zhao et al. 2020] for the same purpose.

As presented, various studies have addressed the teaching of accessibility in higher education computing programs. While these studies have made valuable contributions to the field, many have focused on proposing and evaluating methods without providing sufficient details for other educators to implement these approaches effectively. This work presents preliminary results that suggest a potentially effective and reproducible approach while also including a complete method, an application manual, and slide presentations designed to support instructors in applying it within their own educational contexts. We hope that by providing all the necessary reproduction materials, more programs will be encouraged and better equipped to integrate accessibility teaching into their curricula.

3. Method

3.1. Development of the Method

Project-based Learning (PBL) was the chosen underlying pedagogical approach to base the teaching method. PBL is an instructional approach that immerses students in authentic, real-world projects to foster learning and skill development. It is based on the constructivist finding that students gain a deeper understanding of material when they actively build their understanding by working with and using ideas [Krajcik e Blumenfeld 2005]. Previous research has provided evidence supporting the educational benefits of PBL. Specifically, studies have indicated that higher education students in PBL classrooms perform better than students in traditional instructional environments [Barak e Dori 2005, Alsamani e Daif-Allah 2016, Mohamadi 2018].

With this project-oriented context defined, Design Thinking (DT) emerges as an excellent option for using as a framework to guide the execution of the proposed method. DT, a human-centered problem-solving framework, defines steps to experiment, create and prototype models, gather feedback on them, and redesign them if needed [Razzouk e Shute 2012]. This approach, with its focus on the needs and experiences of the end-users, fosters a sense of connection and empathy, which is crucial in the accessibility educational context. Developing an accessible software problem following the DT process essentially links both of our main objectives: teaching the students to put the needs of the end-users first and constructing the learning process with the development of a real-world project as proposed by PBL.

The method was then developed around the idea of the student following the stages of DT to design an accessible mobile software project. At the beginning of each stage, the essential principles for the development of that stage are presented to the student in the form of a short lecture. Then, the students are encouraged to develop that stage of the DT process in collaborative groups. At the end, the groups are encouraged to share what was developed with the rest of the class. This collaborative approach allows the students to gain theoretical and practical experience and promotes peer learning, enabling them to build upon their own knowledge regarding the development of accessible software.

We specifically designed the method to be used as a module within an existing course rather than as a standalone subject. This approach increases the likelihood

of reaching more students, including those who may not already be interested in accessibility, by embedding the topic into the regular curriculum. While standalone courses can provide deeper exploration, they often suffer from low enrollment and tend to attract only students who already value accessibility. In contrast, integrating the content into existing courses helps ensure broader exposure and promotes a more inclusive educational experience [Putnam et al. 2016].

The definition of the theoretical content presented in the lectures, the development of practical activities, the slide presentations, and the application manual were all based on materials previously used by a professor experienced in teaching accessibility. This professor, who teaches a Human-Computer Interaction (HCI) course, also supervised the development of all activities to ensure their pedagogical soundness. Their participation was essential to guarantee that the structure and content of the method were grounded in proven teaching practices and aligned with the intended learning outcomes.

Once the subjects of the lectures, the initial slide presentations, and the student activities were established, a pilot application of the method was conducted. The primary objectives of this pilot run were to assess the method's effectiveness in improving students' overall understanding of accessibility and to determine the time required for the execution of each activity and stage of the DT process.

Following the execution of the pilot study, which demonstrated the effectiveness of the method in teaching students about accessibility, we considered the varying time constraints of different educational contexts and the need to ensure the adaptability of the method to different time durations. To achieve this, we categorized the activities as required or optional and developed new short and long versions of select activities. Drawing from the insights gained during the pilot study, we also established the minimum time required for each activity.

In this way, various combinations of required and optional activities, as well as short and long versions of each activity, could be chosen, ensuring the method's adaptation to the available time frame. Importantly, this process was conducted under the supervision of the accessibility experienced professor who oversaw the development of the pilot activities. This structured approach resulted in the definition of a shortest possible version of the method, comprising only the shortest version of the required activities, which needs a total of 200 minutes for execution. Conversely, the longest possible version, containing all activities in their longest versions, would require 800 minutes to complete. Many other combinations of activities could be created with durations in between the shortest and the longest, depending on the availability of time for the application of the method.

We conducted testing on two varying durations of the proposed teaching method with two volunteer professors who implemented it in two different courses. This was done to assess the method's effectiveness once again and with different durations while also evaluating the professors' experience with the method and the provided support materials.

3.2. Participants and Ethical Considerations

All the students and professors that participated in this research were sampled by convenience [Golzar et al. 2022]. The students were enrolled in the courses that were used for the execution of the pilot or the two test executions of the method. Students were given a choice to participate or not in the research, and the ones who opted out of the

research still participated in the classes but did not have any data collected or analyzed. The professors were invited to participate and, willingly and without compensation, agreed to participate.

Individuals who agreed to participate were administered a digital informed consent form (ICF). The ICF was distributed electronically a week before the start of the classes so they had time to review and understand it. The research was fully explained to the students and professors before the start of the classes, clarifying all of its ethical considerations and how their data would be handled. Moreover, it was crucial to clarify to students that their participation in the research would not impact their grades or evaluation in the course where the method was applied. This clarification was necessary to remove any bias that students might develop toward wrongfully evaluating the method positively, and to prevent any undue influence on students' decisions to participate.

A total of 43 students participated in the pilot, 11 graduate students and 32 undergraduates. However, only 33 students voluntarily opted to participate in the study and have their data collected. The participants represented diverse academic backgrounds, including four doctoral students in Computer Science, four master's students in Computer Science, 15 undergraduate students in Information Systems, five undergraduate students in Computer Engineering, and five undergraduate students in Computer Science.

The sample size of participants for the two test executions of the method was significantly smaller. Although the courses were full, and all students participated in the method's execution, only five students for the first execution and six students for the second execution chose to have their data collected for the research. The two test executions also had two volunteer professors as participants, each of whom implemented the methods with the students in one course. This research involved human participants and, therefore, was approved by the Universidade Federal do Pará's Committee on Research Ethics under the Proof of Application for Ethical Review number 5.222.718.

3.3. Measurement of Accessibility Awareness

Many studies have proposed different approaches to measure the effectiveness of accessibility teaching methods. While some approaches are based on the qualitative analysis of open-ended questions [El-Glaly 2020] or objective questions about the student's opinions on how the applied teaching method has changed their perceptions of accessibility issues [Shinohara et al. 2017], in [Palan et al. 2017] an Accessibility Awareness measuring technique consisting of a questionnaire through which the participants received an "accessibility awareness score" according to their answers was proposed. Higher composite scores indicated greater accessibility awareness.

For this study, we created an adaptation of the method proposed in [Palan et al. 2017], which also consisted of a questionnaire but was composed of six questions categorized into four learning objectives: overall accessibility awareness, technical knowledge, empathy, and potential endeavors. In [Baker et al. 2020] a systematic literature review of papers on accessibility in computing education was performed and identified these four categories as the key learning objectives commonly covered. We then created questions for each category that aimed to subjectively explore the participants' perceptions of the issue.

Table 1 presents the questions and their categories. Most questions could be

answered on a five-level Likert scale ranging from Strongly Disagree to Strongly Agree, except for questions 3 and 4. Question 4 was a Yes or No question. Participants could receive a score from 0 to 4 points for each answer. Yes or No questions scored 0 for No and 4 for Yes, while Likert-scale questions started at 0 for the most negative response and increased to 4 for the most positive. The total possible score for the questionnaire ranged from 0 to 88 points. This questionnaire was applied twice: once before the start of the method and once after its application, allowing for comparison of scores to assess how the teaching method influenced students' awareness of accessibility.

Table 1. Awareness questionnaire questions and their categories.

#	Category	Question
1	Empathy	I consider it important to include accessibility concerns in software development.
2		I think about the accessibility implications of the software projects I develop.
3	Technical Knowledge	I feel confident and capable of developing accessible software for the following accessibility categories. Answered once for each: low vision, blindness, deafness, autism, learning disability, low literacy, motor disability, older adults and intellectual disability.
4	Overall Awareness	Do you know how people with the following accessibility characteristics interact with computers, cell phones, and other computing devices? Answered once for each: low vision, blindness, deafness, autism, learning disability, low literacy, motor disability, older adults and intellectual disability.
5		I believe teaching accessible software development is important in the curriculum of computing courses.
6	Potential Endeavors	I have a personal interest in developing software products that have accessibility concerns.

3.4. Data Collection and Analysis

In the pilot study, data was collected exclusively from students, as members of the research group applied the method rather than volunteer professors. The participating students were required to complete the accessibility awareness questionnaire twice: once before and once after the implementation of the method. The accessibility scores of the students before and after the application were statistically analyzed to assess the effectiveness of the method in improving their accessibility awareness.

After the pilot, in addition to the accessibility awareness questionnaires answered by participating students, another questionnaire was administered to explore their opinions. This opinion questionnaire aimed to investigate students' perspectives on the teaching method and its impact on their intentions regarding accessible software development. The questions that made up this questionnaire are presented in Table 2. Although the responses to this questionnaire did not affect the students' accessibility

awareness score, they served as a valuable means to explore the students' perspectives on their participation in the experiment.

Table 2. Opinion questionnaire questions.

#	Question
1	The teaching method used sparked my curiosity about the subject of accessibility in software.
2	The teaching method used was interesting and held my attention.
3	After the classes, I believe I know more about the subject of accessibility in software.
4	After classes, I am more likely to consider accessibility in the projects I develop.
5	After classes, I am more likely to consider accessibility as a field of research that I would develop.
6	After the classes, I am interested in deepening my knowledge on the subject of accessibility in software.
7	After classes, I feel more sensitive to the difficulties faced by people who need accessibility.
8	I believe that accessibility in software should be a separate course offered in the curriculum of computing programs.
9	What do you think could have been added to the classes to improve them?
10	Use this space to add criticism, compliment, observations, or any other comments you want about the accessibility classes taught.

While we continued to use the same accessibility awareness questionnaires to assess students during the two subsequent applications of the method, no statistical analysis was conducted due to the smaller sample size. Consequently, the student data is presented only in a quantitative, descriptive form. Although no statistical tests were conducted, the numerical results still offer some insights into the method's potential impact on student awareness. More importantly, this phase focused on evaluating the experiences of the volunteer professors who implemented the method. After its application, we conducted individual interviews with each professor to understand their perspectives. These interviews included 13 structured questions designed to gather detailed feedback on various aspects of the method, as well as suggestions for improvement. Table 3 lists the questions used in the interviews.

All data collected and analyzed in this study is available in Section 7.1.

3.5. Application

3.5.1. Pilot

The pilot was executed in an HCI Course and had a time frame of four classes, each lasting 200 minutes. Members of the research group took turns applying the method and observing the execution to guide us later when adjusting the activities for different time durations; this included understanding the necessary time for each activity, how much

Table 3. Questions for the interview with the professors.

#	Category	Question
1	Previous experiences	Before applying the proposed method, had you already given classes on accessibility or accessible software development?
2	Experience applying proposed method	How was your experience applying the proposed method?
3		Did the material provided provide clear enough guidance for implementing the method in the classroom?
4		Would you adopt the proposed method in a class that you teach in computing courses, why?
5		What were the main difficulties encountered in applying the proposed method?
6	Application manual	How do you evaluate the method application manual?
7		Did you miss the explanation of any concept in the application manual that would help you in the application of the classes?
8		Based on your experience, do you have any suggestions for improving the application manual?
9	Slide presentations	How do you evaluate the slides for the presentation of the classes
10		Based on your experience, do you have any suggestions for improving the slides?
11	Feedback and suggestions	Based on your experience, do you have any suggestions for improving the proposed method?
12		Is there any other information you would like to share about your experience with the proposed teaching method?
13		Would you recommend this method to other professors interested in including accessibility teaching in the subjects they teach?

time each DT stage should take of the total time, and what activities could have versions with reduced or increased duration.

3.5.2. Tests

We enlisted two volunteer professors to teach different versions of our method for varying time durations. We asked them how much time they could dedicate to the method during their courses, and based on their response, we customized the method's activities to fit within their time constraints. We also created personalized slide presentations and application manuals based on the activity division that could be used to guide the application of the method. The instructions for tailoring the method to specific time frames are provided with the supplemental material (see section 7.1).

Professor 1 implemented a version of the method with three meetings, each lasting 100 minutes, for a Software Engineering course that will be referred to as Course 1. On the other hand, Professor 2 conducted the shortest possible version of the method,

comprising two meetings of 100 minutes each, for a Software Engineering Laboratory class, referred to as Course 2.

4. Accessibility Teaching Method

The execution of the proposed teaching method focuses on students designing a user-friendly mobile application for a specific accessibility need. At the beginning of the method's execution, students are divided into groups, and each group is asked to choose an area of accessibility for which to develop their app, such as blindness, older adults, or deafness. The stages of the DT process are then followed, advancing with the design in each stage. Each stage comprises multiple activities, starting with a lecture to introduce the students to the essential concepts used during that stage. After the activities have been developed, each stage ends with the group presenting to the class what was developed.

Table 4 presents a complete description of the proposed method. It details all the activities that make up the method, including their descriptions and corresponding DT stages. The column "Rqd." indicates whether an activity is required, these activities are essential and must be included in any implementation of the method. The column "Ver." specifies the version of the activity, classified as long or short, to support adaptation to different time constraints. If the version is marked as "n/a", the activity has only one form. Finally, the "Minimal duration" column shows the minimum time needed to carry out each activity. The application manual, the slide presentations, and information on adapting a version of the method for specific time durations are included in the paper's supplemental material (see section 7.1).

4.1. Design Thinking Stages and Activities

Even though Design Thinking is an iterative process, for this accessibility teaching method, only a single linear iteration is conducted. Each stage of the process was designed to achieve a specific learning objective and to build upon the knowledge gained in previous stages. By the end of the iteration, the students will have been introduced to various important topics related to different phases of the software development process.

- **Empathize:** The goal is to introduce students to various accessibility characteristics and raise awareness of the diverse needs of individuals. This stage emphasizes the importance of considering these needs when developing software.
- **Define:** Students are introduced to tools that help adapt software for accessibility. The focus is on understanding and applying the WCAG guidelines to address specific needs, enabling them to define requirements for accessible software design.
- **Ideate:** Students begin envisioning the mobile application they will create, combining concepts from previous stages. They define the app's purpose and how to make it accessible for their assigned user group.
- **Prototype:** Students develop low-fidelity prototypes using paper or prototyping tools. The objective is to apply learned concepts and visibly demonstrate how the design addresses the chosen accessibility requirements.
- **Test:** Students conduct paper prototype tests to simulate app interaction. Testing is preferably done by members of other groups to highlight the iterative nature of design and the importance of feedback. Testers are encouraged to discuss issues, prompting groups to refine their solutions.

Table 4. Activities of the proposed teaching method.

Stage	Activity	Rqd.	Ver.	Minimal Duration	Description
Empathize	Learning about accessibility characteristics	Yes	Short	20 min.	The professor presents and divides the characteristics between each group. Lecture explaining the needs of each one, and how they interact with computers.
			Long	30 min.	Each group chooses an accessibility characteristic and must use the internet to better understand the chosen characteristic.
	Present to the class	No	n/a	30 min.	Each group must present to the class what they learned about the chosen accessibility characteristic.
Define	Learning about accessibility and guidelines	Yes	Short	20 min.	Lecture presenting the legal and formal concepts of accessibility, and also presenting the concept of guidelines, focusing on WCAG.
			Long	40 min.	Lecture presenting the concepts of accessibility, and guidelines. Interaction evaluating interfaces to identify accessibility problems.
	Guideline association	No	n/a	20 min.	Look at the WCAG, and select the criteria that should be followed for people with the group's accessibility characteristic.
	Present to the class	No	n/a	30 min.	Each group should present to the class the guidelines identified in the guideline association activity.
Ideate	Learning about accessibility in software development	Yes	n/a	15 min.	Lecture on how accessibility should be embedded in all stages of software development.
	Idealize the application	Yes	n/a	30 min.	Each group must define the key points of the application that they will prototype, such as its purpose and what measures will be taken to ensure accessibility to the target audience.
	Present to the class	No	n/a	30 min.	Each group must present to the class everything that was defined for its application.
Prototype	Learning about prototyping	Yes	n/a	15 min.	Lecture on the importance of prototyping, and also on the different ways that software can be prototyped. Including a paper prototype, which will be what the groups will develop.
	Accessible development in practice	No	n/a	30 min.	Lecture with practical examples of accessibility being inserted into code in different languages and frameworks.
	Prototype the application	Yes	Short	30 min.	Development of the paper prototype of the planned application.
			Long	60 min.	
	Present to the class	No	n/a	30 min.	Each group should present the developed prototype to the class.
Test	Learning about testing	Yes	n/a	15 min.	Lecture on tests, types of tests, concept and persona and test scenarios. Focus on carrying out tests with a paper prototype.
	Test the application	Yes	Short	30 min.	Definition of the test scenario and persona and testing of the prototype.
			Long	60 min.	
	Present to the class	No	n/a	30 min.	Each group should present their test findings to the class.

5. Results

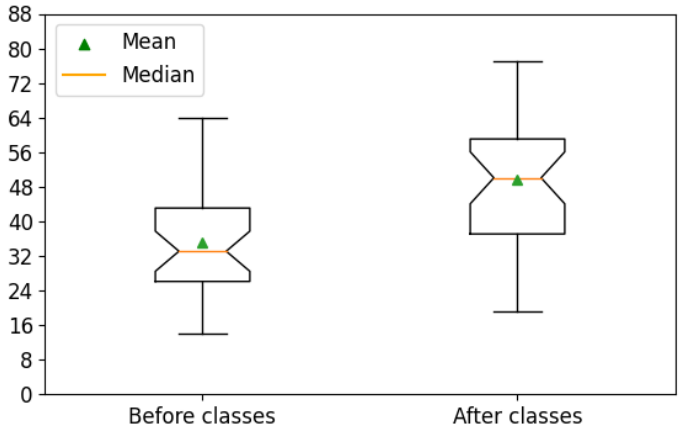
The students' accessibility awareness scores collected before and after the method applications were analyzed statistically and quantitatively. This section also presents the results of the interviews with the volunteer professors who applied Courses 1 and 2.

5.1. Pilot with Students

5.1.1. Accessibility Awareness

The students' awareness scores were considerably higher in the questionnaire applied after classes ($Mean = 49.7, Median = 50, StDev = 14.7$) than in the questionnaire applied before classes ($Mean = 35, Median = 33, StDev = 11.4$). A box plot comparing the students' awareness scores before and after the classes is shown in Figure 1.

Figure 1. Distribution of students' awareness scores before and after classes.



The Student's t-test for dependent samples was applied based on that analysis, given that our sets met all other assumptions for this test. This parametric test is used when comparing repeated measurements on the same subjects before and after an intervention, and it shows how significant the differences between the sets' means are. The results ($t = 5.31, p = 0.000008$) allowed for the rejection of the test's null hypothesis, as $p < \alpha$ for $\alpha = 0.05$, which shows that the increase in the student's awareness scores was statistically significant and that the teaching method applied was successful in increasing their accessibility awareness scores.

5.1.2. Opinion Questionnaire

After classes, the participant students also answered a questionnaire that sought to explore their opinions about the applied teaching method. This questionnaire had eight questions on a 5-point Likert scale, where students could answer each question from 1 to 5, for Strongly Disagree and Strongly Agree, respectively, and two open-ended questions.

In the first question, the students answered how much they agreed that the applied teaching method sparked their curiosity about the topic of accessibility. Of

the 33 respondents, 69.7% (23) answered 5 on the Likert scale, indicating a strong agreement with the statement. Moreover, 24.2% answered 4, and 6.1% answered 3. The second question explored whether students agreed that the applied teaching method was interesting and held their attention. Of the 33 respondents, 51.5% (17) answered 5 on the Likert scale, indicating they strongly agree with the statement, and 42.4% (14) answered 4. The rest of the students (6.1%) answered 3.

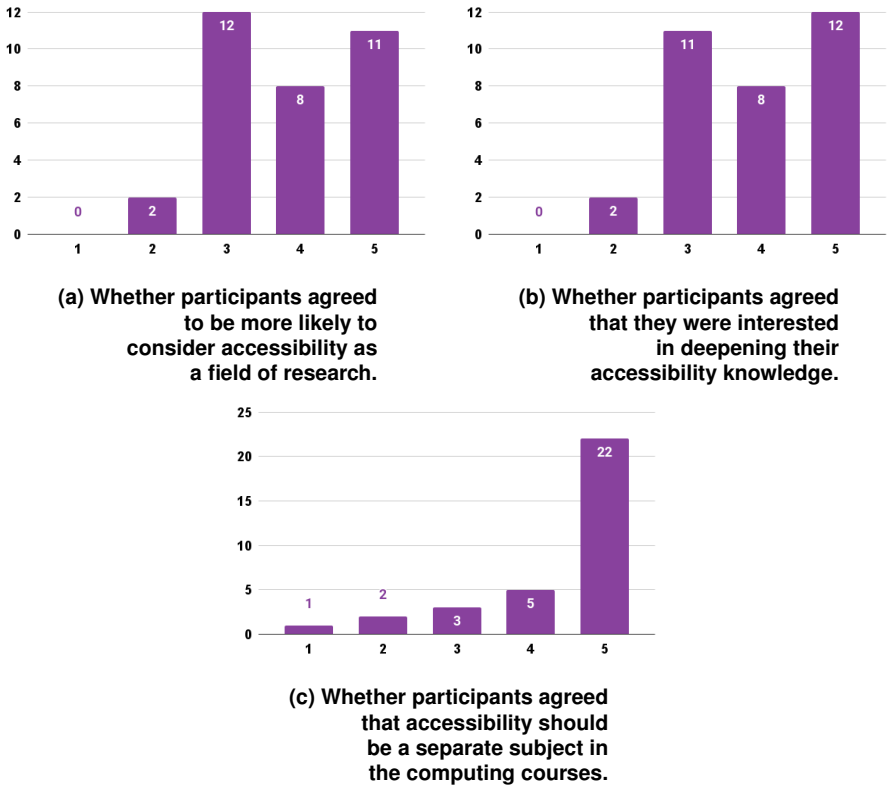
We explored with the third question whether students believed they knew more about accessibility after the classes. Of the 33 respondents, 51.5% (17) answered 5 on the Likert scale, indicating they strongly agree with the statement, and 36.4% (12) answered 4. The rest of the students (12.1%) answered 3. When asked in the fourth question if they were more likely to consider accessibility in the software projects they develop in the future, 72.7% (24) of the students answered 5 on the Likert scale, indicating they strongly agree with the statement. Moreover, 21.2% responded 4, and 6.1% responded 3.

Responses to the fifth question, which explored whether students agreed that after the classes, they were more likely to consider accessibility as a field of research, were more widespread, as shown in Figure 2a, with the majority of participants (36.4%) answering 3. The sixth question explored whether students agreed that they were interested in deepening their accessibility knowledge after classes. Figure 2b shows the distribution of answers, with the majority of participants (36.4%) answering 5 on the Likert scale.

For the seventh question, the students answered how much they agreed they are more sensitive to the difficulties faced by people who need accessibility after classes. Of the 33 respondents, 75.8% (25) answered 5 on the Likert scale, indicating a strong agreement with the statement. Moreover, 18.2% answered 4, and 6.1% answered 3. The eighth question explored whether participants agreed that software accessibility should be a separate subject offered in the curriculum of computing courses. Most participants (66.7%) answered 5 on the Likert scale, indicating a strong agreement with the statement. However, the rest of the answers were widespread, as shown in Figure 2c.

The ninth question was an open-ended and non-mandatory question where students could expose what they thought could have been added to the classes to improve them. Only ten students answered this question, and five indicated that they missed direct contact with a person with a disability who could report more about their first-hand experiences to the class. Two students answered that they would have liked to have a moment in a laboratory to test accessibility software, such as screen readers, to better understand how it works. The other three students answered that they would like the lecture moments to explain in more depth about accessibility guidelines, prototyping techniques, and more examples of designs that meet or violate accessibility guidelines.

Finally, the tenth question was also open and non-mandatory where students could present criticism, compliments, observations, or any other comment they wanted about the accessibility classes. Only eight students answered this question. All responses complimented the initiative and the methodology employed, recognizing the importance of teaching accessibility. One of the students also added, “[...] (the classes) brought me more empathy about the topic addressed, now every time I start prototyping something I already think about how I can make it more accessible”.



(a) Whether participants agreed to be more likely to consider accessibility as a field of research.

(b) Whether participants agreed that they were interested in deepening their accessibility knowledge.

(c) Whether participants agreed that accessibility should be a separate subject in the computing courses.

Figure 2. Answers to the fifth, sixth, and eighth questions of the opinion questionnaire. Likert scale, 1 for Strongly Disagree, 5 for Strongly Agree.

5.2. Courses 1 and 2 with Students and Professors

Professor 1 applied the method to a Software Engineering course with 15 undergraduate students in Information Systems. Course 1 consisted of three meetings of 100 minutes. Considering the parameters, Professor 1 had 5 hours to execute the method. Following the tutorial to create an adapted version of the method that can be found in the supplemental material (see section 7.1), the activities for Course 1 were allocated in each stage.

Despite not being included in the initial adaptation, Professor 1 managed to fit the execution of all "Present to the class" activities in Course 1, taking advantage of the fact that they had few groups and could make their execution shorter. Using his own experience, Professor 1 chose to add to Course 1 the activities "Guideline association" and "Accessible development in practice". Educators are encouraged to adapt the method to their context and their experience. Table 5 shows the activities performed in Course 1, with the extra activities added by the professor marked by an asterisk and in italics.

Professor 2 applied the proposed method to a Software Engineering Laboratory course, also with 15 undergraduate students in Information Systems. Professor 2 had at their disposal only two meetings, each 100 minutes long. Considering the parameters, Professor 2 had a total of 3:20h to execute the method. Course 2 received the execution of the shortest possible version of the proposed method, and Professor 2 applied to the students only the required activities, all in their shortest versions, as shown in Table 6.

Figures 3a and 3b present box plots of the distribution of students' awareness scores before and after the implementation of the method for Courses 1 and 2,

Table 5. Activities of the method for Course 1.

Stage	Activity
Empathize	Learning about accessibility characteristics - Short
	*Present to the class**
Define	Learning about accessibility and guidelines - Short
	*Guideline association**
	*Present to the class**
Ideate	Learning about accessibility in software development
	Idealize the application
	*Present to the class**
Prototype	Learning about prototyping
	*Accessible development in practice**
	Prototype the application - Short
	*Present to the class**
Test	Learning about testing
	Test the application - Short
	*Present to the class**

Table 6. Activities of the method for Course 2.

Stage	Activity
Empathize	Learning about accessibility characteristics - Short
Define	Learning about accessibility and guidelines - Short
Ideate	Learning about accessibility in software development
	Idealize the application
Prototype	Learning about prototyping
	Prototype the application - Short
Test	Learning about testing
	Test the application - Short

respectively. Analyzing the box plots reveals a substantial increase in students' awareness scores following the application of the method in both cases. This noticeable improvement is reflected in the elevated values of both the mean and median values. Furthermore, an observable positive shift in the samples' overall distribution further underscores the method's efficacy. To provide a more granular observation of students' awareness scores, Figures 4a and 4b illustrate their individual scores before and after the application of the method for Courses 1 and 2, respectively.

5.2.1. Interviews with Professors

Interviews were conducted with Professor 1 and Professor 2 after completing the method's application to evaluate the professors' viewpoints. The interview questions, shown Table 3, aimed to explore their opinions and experiences with the proposed method. Subsequently, their responses will be presented, organized by similar themes, and accompanied by excerpts that reflect their opinions.

Previous experiences: As for the professors' previous experience with teaching accessibility, both stated that they had covered accessibility topics in classes before.

Figure 3. Distribution of students' awareness scores before and after the classes.

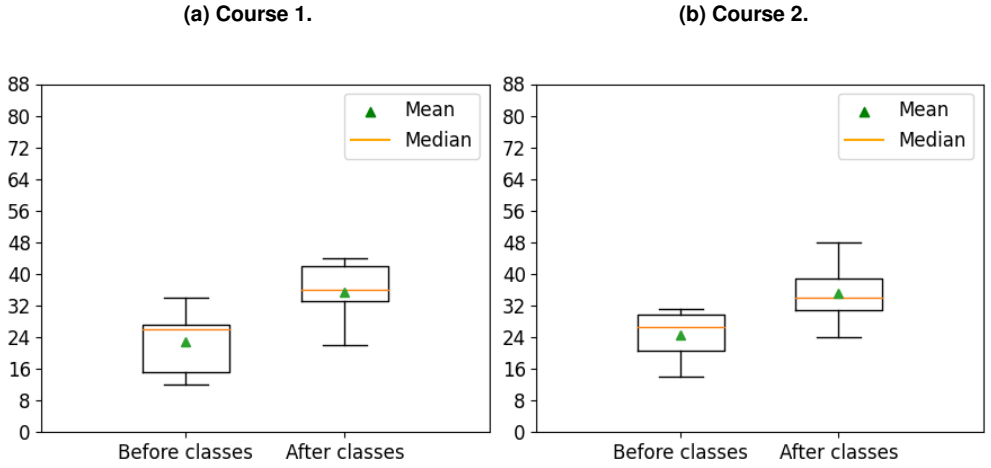
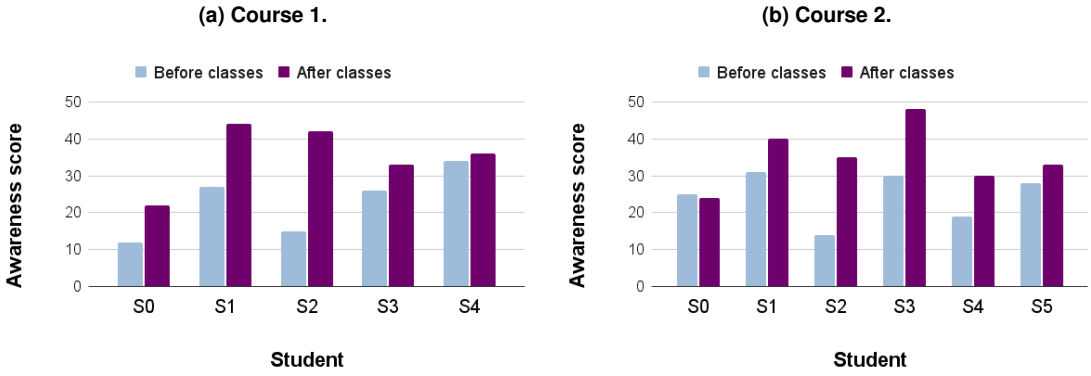


Figure 4. Students individual awareness scores evolution.



Professor 1 had taught about accessibility in more detail in a previous course, including teaching about accessibility in software development. On the other hand, Professor 2 mentioned that they had touched upon accessibility in software to some extent in the past, but not as extensively as the proposed method suggests.

Experience applying the proposed method: Both professors expressed their positive feedback on applying the method. Professor 1 stated, "It was very cool, we managed to develop a lot with the class and pass on these concepts well, I believe they were able to absorb [...]". Professor 2 also praised the method and suggested, "[...] I would suggest perhaps optimizing a few more things, trying to work the practical and theoretical parts together to [...] save a little more time". Notably, Professor 2 used the shortest version of the method, completing the application in only 200 minutes.

We asked if the material offered sufficient guidance for implementing the method. Both professors agreed that the material was clear and provided sufficient guidance for conducting the activities. However, they also offered some suggestions for improvement.

Professor 1 mentioned that the application manual could be simplified in terms of its size and amount of text but emphasized that this was just a minor suggestion and did not pose any significant issues, "[...] It's just a suggestion, nothing that interfered." Professor 2 suggested having a shorter version of the material, particularly for the slide presentations in the shortest version of the method, noting, "[...] perhaps in the shorter version try to have a shorter material. Try to, somehow, make it shorter for a shorter class time.".

Both professors indicated some difficulties with the organization of time. Professor 1 mentioned challenges in managing the time for activities and expressed concerns about whether the time was sufficient to complete everything during class. Professor 2, using the shortest version of the method, suggested giving more emphasis to practical activities over theoretical ones. Despite these challenges, both professors responded that they would adopt the method to apply in future computing classes they teach. Professor 1 stated, "Certainly, I think you can adopt it and I would adopt it. Because it's important, it's interesting and it motivates the class well [...]". Professor 2 emphasized the significance of accessibility knowledge, saying, "Yes, because I think it is very important for us to have this knowledge of accessibility [...]".

Application manual: Both professors evaluated the application manual positively. Professor 1 stated, "[...] it is very explanatory, it manages to describe well the activities that we are going to do and that are proposed and it manages to make us understand how to put it into practice [...]". Professor 2 commented, "I think the manual is very complete and matches the slides very well, explaining well each point, how to work and gives you a time dimension of how to work each subject [...]". Both professors found the manual sufficient to explain the necessary concepts for their classes. However, Teacher 1 suggested making the manual more visually oriented to allow teachers to personalize it more easily and with less reliance on textual content.

Slide presentations: The professors also had positive feedback for the slide presentations provided. Professor 1 mentioned that the slides effectively guided the classes, and Professor 2 appreciated their continuity between activities, making them an excellent guide for the classes. Nevertheless, Professor 1 suggested some improvements, saying, "[...] I missed having more images to better illustrate and exemplify, and also a more direct definition of the stages of design thinking and what that means [...] in the parts of the code examples to have videos of the technologies running would be nice." Professor 2 recommended merging some slides to be more concise and focused, stating, "[...] I think that with the short time I could reduce and dry up some things, things from 3 slides become 1 or 2 to be more objective and focus the student on the practical part.".

Feedback and suggestions: Regarding the method, both professors provided constructive suggestions for improvement. Professor 1 proposed enhancing the professor's ability to customize the method within each version, with the support material offering more flexibility. The professor explained, "[...] what I can suggest is that it could allow for better customization [...] if the text could support the professor in personalization, it would be cool. Even on the slides, put optional things for the professor to remove or include.". Professor 2's only suggestion was to provide a deeper explanation of the WCAG and its adherence levels. The professor pointed out, "Perhaps the WCAG levels could be clearer, make it a little clearer for students with more practical and

common examples.".

Finally, both educators agreed when asked if they would recommend the proposed method to other professors. Professor 1 stated, "Yes, I would recommend it. I think it is very useful and for those who have never talked about accessibility in class, it is very important because it is a guide, right? It helps a lot and I would recommend it.". Professor 2 commented, "Yes, I think this method combines well with some disciplines, from disciplines focused on eng soft, but also programming disciplines themselves, it gives the student a sense of how accessibility issues should be worked within a software and especially within the life cycle of a software.".

6. Discussion

The study results demonstrate that the proposed method successfully enhanced students' knowledge, awareness, and empathy toward accessibility. This finding is supported by the statistical significance of the results of the pilot application and the consistent increase in awareness scores across all applications, regardless of duration. The results align with prior research emphasizing the value of hands-on, user-centered approaches in fostering inclusive design practices [Altay et al. 2016]. For instance, the statistically significant increase in awareness scores (mean improvement of 14.7 points) mirrors findings from [Zhao et al. 2020], where project-based interventions performed better than traditional lectures.

Students' perceptions of the method were investigated through a post-class questionnaire aimed at capturing their opinions and intentions. The majority reported that the method increased their accessibility knowledge, sparked curiosity, and sustained their engagement during the sessions. Many also indicated a greater likelihood of considering accessibility in future projects, heightened sensitivity to accessibility-related challenges, and a belief that accessibility should be a standalone subject in computing curricula. This feedback was crucial to assess students' overall satisfaction, engagement, and the extent to which the experience may influence their future design practices. These results also confirm the expectations shared by 12 professors in [Putnam et al. 2016], where the most common takeaway when teaching classes about software accessibility was the desire for students to develop empathy and embrace user diversity in their careers.

The professors' perspectives gathered from the interviews reveal a positive experience applying the method and a positive evaluation of the support material provided. This positive feedback from the professors indicates that the method is not only effective for students but also helpful and practical for educators. The professors' experiences implementing the accessibility teaching method reveal insights into the balance between pedagogical structure and adaptability. While the method's modular design was praised for its practicality, the feedback underscores the challenge of balancing standardized frameworks with the diverse realities of real-life classrooms.

Professor 1 emphasized the importance of making the method more customizable for educators, such as making it clear in the application manual and on the slides which parts could be removed or included at their discretion. Currently, the only suggestion given related to this kind of personalization is that the professors have the autonomy to edit the slides, add relevant content, or add new activities as long as they fit within the available time. However, further investigation is needed to identify specific moments

where customization can be applied without compromising the method's effectiveness.

Professor 2 implemented the shortest version of the method, which consists of only 200 minutes, and highlighted the difficulty in managing time to cover all activities within the limited timeframe. They suggested the creation of more concise support material for this version or even a stronger focus on practical activities over theoretical ones. Further research is needed to determine if such adjustments can maintain the method's effectiveness. Alternatively, based on Professor 2's experience, the shorter version of the method might be more effective if extended to at least 300 minutes, as applied by Professor 1, rather than just 200 minutes.

The feedback from professors highlights that a single-method proposal may not be suitable for every scenario, no matter how adaptable you conceive it. Derivations of the method might be needed for extremely short scenarios or even scenarios where the professor already applies a class-long project that would be interrupted by the proposed method's application.

Unlike all the other similar proposals mentioned in the Section 2, which usually describe general ideas or examples of activities, our proposal goes further by providing professors everything needed to apply the method. The materials include a detailed guide, editable slides, and ready-made activities that can be used or adapted as needed. This is one of the main contributions of our work. By offering these concrete materials, we help make it easier for educators to include accessibility in their classes, even if they do not have much experience with the topic. Our hope is that by offering ready-to-use materials, we can make it easier for educators to teach accessibility in their classrooms and encourage the inclusion of this topic more broadly within computing curricula.

While the findings are encouraging, it is important to acknowledge that they are preliminary. The study was limited by the number of applications of the method, the relatively small sample sizes of the two last applications, and the use of convenience sampling. These factors constrain the generalizability of the results. Additionally, the absence of long-term follow-up means we cannot yet assess whether increased awareness and empathy lead to sustained behavioral change in professional practice. Further research is necessary to replicate the method in different institutional contexts, with varied student and course formats, to better understand its scalability, adaptability, and long-term impact.

Approximately 78% of students strongly agreed that they felt more sensitive to accessibility challenges after participating in the pilot execution of the method. However, although empathy can be a strong motivator, and this increase addresses one of the root causes of non-accessible design, its translation to real-world practices depends on many other factors [Silva et al. 2022]. This proposal alone will not solve the persistent gaps in accessibility education and practice. Rather, it represents just one piece of a larger, systemic issue that must be addressed from multiple fronts. Institutional commitment to accessibility, integration of accessibility content across the curriculum, availability of inclusive design tools, adjustment of industry practices and supporting legislation are all critical components. Without structural changes in how accessibility is valued and assessed, both in academia and in software development pipelines, educational interventions may raise awareness but struggle to sustain long-term change.

7. Final Considerations

This study presented the proposal of a method for teaching accessibility to students in higher education computing courses. The effectiveness of the method was evaluated through the execution of three different tests with varying durations, demonstrating its value as a learning tool for students. The professors' perspectives were also considered, involving two volunteer professors who implemented two versions of the method in Information Systems classes. The professors viewed the method as effective, expressing interest in using it again in the future and recommending it to other professors. They also provided valuable feedback for potential improvements in future iterations.

7.1. Reproducibility

In order to ensure that the method can be easily replicated, the supplementary material[1] comprises all the necessary support resources. These resources include a manual that provides detailed instructions for implementing the activities, slide presentations for each class that outline the required theoretical knowledge and provide step-by-step guidance for each activity, as well as a tutorial on modifying the method to suit different time constraints. The adaptation tutorial includes instructions for adjusting the application manual and slide presentations to include only those activities that can be accommodated within a specific time frame. The material is available in Portuguese and English.

All of the data collected through questionnaires and interviews during the development of this research is also available at the same repository linked above. The data is presented in Portuguese and includes responses gathered from both students and professors who participated in the study.

7.2. Future Work

As future work, we plan to incorporate an evaluation strategy into the proposed accessibility teaching method. This addition aims to provide professors adopting the method with a means to assess their student's participation in the classes. This will allow professors to fully incorporate the proposed method as part of their classes and as part of the student evaluation for the duration of the course.

Future work also includes conducting new evaluations of the method with professors from diverse experience backgrounds, as well as with students from different courses and semesters, and exploring varying durations of method implementation. We also intend to include qualitative research methods in these evaluations, such as interviews and focus groups. These investigations will provide valuable insights into the method's effectiveness across different execution scenarios and contribute to a deeper understanding of its adaptability and impact in a range of educational contexts.

Acknowledgement

Language tools were used in the writing of this paper to assist in adapting and proofreading the text for clarity and accuracy in the English language. These tools may include AI-driven grammatical correction and translation.

We wish to acknowledge Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) for funding and supporting the development of this study.

[1] Available at: `https://github.com/AlanTas/IHC2025_Accessibiliy_Teaching`

References

Alsamani, A.-A. S. e Daif-Allah, A. S. (2016). Introducing project-based instruction in the saudi esp classroom: A study in qassim university. *English Language Teaching*, 9(1):51–64.

Altay, B., Ballice, G., Bengisu, E., Alkan-Korkmaz, S., e Paykoç, E. (2016). Embracing student experience in inclusive design education through learner-centred instruction. *International Journal of Inclusive Education*, 20(11):1123–1141.

Antonelli, H. L., Rodrigues, S. S., Watanabe, W. M., e de Mattos Fortes, R. P. (2018). A survey on accessibility awareness of brazilian web developers. In *Proceedings of the 8th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion*, pages 71–79.

Baker, C. M., El-Glaly, Y. N., e Shinohara, K. (2020). A systematic analysis of accessibility in computing education research. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, pages 107–113.

Barak, M. e Dori, Y. J. (2005). Enhancing undergraduate students' chemistry understanding through project-based learning in an it environment. *Science education*, 89(1):117–139.

Bi, T., Xia, X., Lo, D., Grundy, J., Zimmermann, T., e Ford, D. (2022). Accessibility in software practice: A practitioner's perspective. *ACM Trans. Softw. Eng. Methodol.*, 31(4).

Bohman, P. R. (2012). *Teaching Accessibility and Design-For-All in the Information and Communication Technology Curriculum: Three Case Studies of Universities in the United States, England, and Austria*. Doctoral dissertation, Utah State University, Logan, UT, USA. All Graduate Theses and Dissertations, 1369.

El-Glaly, Y. N. (2020). Teaching accessibility to software engineering students. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, pages 121–127.

Freire, A. P., Russo, C. M., e Fortes, R. P. M. (2008). A survey on the accessibility awareness of people involved in web development projects in brazil. In *Proceedings of the 2008 International Cross-Disciplinary Conference on Web Accessibility (W4A)*, W4A '08, page 87–96, New York, NY, USA. Association for Computing Machinery.

Gabbert, P. (2020). Teaching accessibility in a cs0 class. *Journal of Computing Sciences in Colleges*, 35(7):11–20.

Golzar, J., Noor, S., e Tajik, O. (2022). Convenience sampling. *International Journal of Education & Language Studies*, 1(2):72–77.

Gomes, B., Rios, J., e Rodrigues, K. R. H. (2020). Challenges for the implementation of accessible web and mobile systems. In Santos, R. P. d., Maciel, C., e Viterbo, J., editors, *Software Ecosystems, Sustainability and Human Values in the Social Web*, pages 138–158, Cham. Springer International Publishing.

Krajcik, J. S. e Blumenfeld, P. C. (2005). *Project-Based Learning*, pages 317—-334. Cambridge Handbooks in Psychology. Cambridge University Press.

Ludi, S., Huenerfauth, M., Hanson, V., Rajendra Palan, N., e Conn, P. (2018). Teaching inclusive thinking to undergraduate students in computing programs. In *Proceedings of the 49th ACM technical symposium on computer science education*, pages 717–722.

Mohamadi, Z. (2018). Comparative effect of project-based learning and electronic project-based learning on the development and sustained development of english idiom knowledge. *Journal of Computing in Higher Education*, 30(2):363–385.

Palan, N. R., Hanson, V. L., Huenerfauth, M., e Ludi, S. (2017). Teaching inclusive thinking in undergraduate computing. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 399–400.

Putnam, C., Dahman, M., Rose, E., Cheng, J., e Bradford, G. (2016). Best practices for teaching accessibility in university classrooms: Cultivating awareness, understanding, and appreciation for diverse users. *ACM Trans. Access. Comput.*, 8(4).

Razzouk, R. e Shute, V. (2012). What is design thinking and why is it important? *Review of educational research*, 82(3):330–348.

Shinohara, K., Bennett, C. L., Wobbrock, J. O., e Pratt, W. (2017). Teaching accessibility in a technology design course. In Smith, B., Borge, M., Mercier, E., e Lim, K., editors, *Making a Difference: Prioritizing Equity and Access in CSCL: 12th International Conference on Computer Supported Collaborative Learning (CSCL) 2017, Volume 1*, pages 239–246. International Society of the Learning Sciences (ISLS).

Silva, A. T. d. A., Teran, L. A., Rocha, T. Á. d., e Mota, M. P. (2022). A study on accessibility aspects of software development in brazil. In *19th CONTECSI 2022 - Proceedings and Abstracts*.

Teran, L. A., Silva, A. T. d. A., Melo, G. L. N., e Mota, M. P. (2021). A study on discovering accessibility issues in the software development process. In *18th CONTECSI 2021 - Proceedings and Abstracts*.

Waller, A., Hanson, V. L., e Sloan, D. (2009). Including accessibility within and beyond undergraduate computing courses. In *Proceedings of the 11th international ACM SIGACCESS conference on Computers and accessibility*, pages 155–162.

Wang, Y. D. (2012). A holistic and pragmatic approach to teaching web accessibility in an undergraduate web design course. In *Proceedings of the 13th annual conference on Information technology education*, pages 55–60.

Zhao, Q., Mande, V., Conn, P., Al-khazraji, S., Shinohara, K., Ludi, S., e Huenerfauth, M. (2020). Comparison of methods for teaching accessibility in university computing courses. In *The 22nd International ACM SIGACCESS Conference on Computers and Accessibility*, pages 1–12.